OOV Words Extension for Modular Neural Acoustics-to-Word Model

Hao Li, Zhehuai Chen, Qi Liu, Yanmin Qian, Kai Yu
SpeechLab, Department of Computer Science and Engineering
Shanghai Jiao Tong University, Shanghai, China

Abstract: The direct Acoustic-to-Word (A2W) End-to-End (E2E) model leads to a truly E2E speech recognition system, which gets rid of lexicons or sub-word models. Nevertheless, one of the challenges hampers the wide application of A2W models is the out-of-vocabulary (OOV) or vocabulary expansion problem. This work tries to solve the OOV problem by extending the adaptability of A2W E2E models. We follow the modular training framework of E2E systems and decouple the A2W E2E model into an acoustics-to-phoneme (A2P) module and a phoneme-to-word (P2W) module before joint optimization. Benefit from the modularization, the OOV modeling techniques can be explicitly applied in the latter part. We utilized the extra text corpus for OOV data augmentation. Words are transcribed into phonemes by a pronunciation lexicon or grapheme-to-phoneme transducer (G2P). After that, the augmented OOV P2W data can be used to fine-tune the pre-trained modular system with the extended vocabulary. To alleviate the augmented data mismatch, several methods of data synthesis and model training are investigated and compared. The proposed method is applied in both CTC and attention-based encoder-decoder models and shows significant and consistent improvements in both Switchboard telephone and WSJ reading speech.

Keywords: E2E, ASR, OOV, cross-domain.

1. introduction

Automatic Speech Recognition (ASR) aims to map acoustic sequences to corresponding word sequences. Due to the stronger modeling ability in the context and history of sequence modeling [1][2] and more labeled data, End-to-end (E2E) ASR system can directly map acoustics to words using a unified model, simplifying the ASR pipeline. Both the connectionist temporal classification (CTC) [3][4] and the attention-based encoder-decoder model [5][6] report promising performance in E2E system. Especially the direct Acoustic-to-Word (A2W) model leads to truly end-to-end speech recognition models, which gets rid of lexicons or sub-word models[1] [7]. Another strong motivation for building A2W is to work on semantically meaningful representations of inference labels [8], which are words in almost all languages. In this work, we focus on A2W E2E speech recognition.

One of the challenges hampers the wide application of A2W models is the out-of-vocabulary (OOV) or vocabulary expansion problem. Due to the fixed vocabulary, A2W cannot predict OOV words. Moreover, if words appear in the training dataset rarely, they will be underfitting even if they are in the vocabulary. Similar problems exist in the vocabulary expansion problem, where hot-words or user-specific entities are hard to be added into the speech recognition engine in the A2W E2E framework. All prior works try to solve this problem by introducing separate grapheme level units into the modeling [3][9]. These works still take E2E models as a whole and do not look into the essential adaptability problem of E2E models. Furthermore, the introduced units result in implicit modeling and alignment complexities, which will be discussed in Section 2.

This work tries to solve the OOV problem by extending the adaptability of A2W E2E models. [10] proposed to decouple the A2W E2E model. In the modular training, an acoustics-to-phoneme model

[1] Sub-word units are unavailable in some languages, e.g. Mandarin.

(A2P) and a phoneme-to-word model (P2W) are trained separately before joint optimization. Benefit from the modularization, the OOV modeling techniques can be explicitly applied in the latter part. To handle the OOV problem in A2W models, we utilized the extra text corpus for OOV data augmentation. Since the P2W module deals with mapping the phoneme sequence to the word sequence, we transcribe words into phonemes by using a pronunciation lexicon or grapheme-to-phoneme transducer (G2P). After that, the augmented OOV P2W data can be used to fine-tune the pre-trained modular system with extended vocabularies. To alleviate the augmented data mismatch, several methods of data synthesis and model training are investigated.

The proposed method is verified in both CTC and attention-based encoder-decoder models and shows significant improvement in Switchboard and WSJ corpus. Experiments show that the proposed method achieves 5% and 20% relatively WER improvements in sentences containing OOV for the in-domain and cross-domain conditions respectively.

The rest of the paper is organized as follows. In Section 2, the prior works are briefly reviewed. In Section 3, the E2E training and modularized framework are reviewed. In Section 4, the OOV data augmentation and fine-tune training strategy are proposed in detail. Experimental results and analysis are conducted on Switchboard and WSJ corpus in Section 5. Finally, we make a conclusion and look forward to future work in Section 6.

2. Relation to prior work

There are many works in the traditional systems to handle OOV problem for open vocabulary ASR task. Some works [11] detect OOV words and recognize their phonetic transcriptions. [12] goes further to identify the character sequence of OOV words using P2W conversion to generate the character sequence. The proposed method also takes phonemes as the mediator between acoustics and words. The P2W conversion is implicitly done by joint optimization in Section 3.

The E2E model is less adaptive, which makes the OOV problem harder. All prior works try to solve this problem by introducing separate grapheme level units into the modeling. [9] proposes to model words and graphemes separately using multi-task learning. Since word and character sequences for an input speech utterance are not guaranteed to be synchronized in time [13], their alignment method needs more investigation. [3] proposes to model words and graphemes in a unified softmax layer. The introduced independent assumption between a certain word and its corresponding graphemes may hurt modeling effects. Overall speaking, all prior works try to solve this problem by introducing separate grapheme level units into the modeling [3][9]. These works still take E2E models as a whole and do not look into the essential adaptability problem of E2E models. Moreover, if the training set is mismatch with the test set, although these methods do not have OOV problem, the rare words still cause confusion to these models. The problem can never be handled by just using the training data.

The proposed method is inspired by recent work in improving E2E systems. To utilize text data, [14] proposes to "back-translate" text to acoustic features, which can be used to train the E2E model. We systematically investigate this method in the OOV problem.

3. Modular Acoustic-to-Word System

3.1 Connectionist temporal classification

Connectionist temporal classification (CTC) [15] provides a direct way to calculate the posterior probability $P(\mathbf{l}\,|\,\mathbf{x})$ of the target sequence $\mathbf{l}$ given the feature sequence $\mathbf{x}$. In order to calculate the sequence posterior, CTC introduces an additional *blank* symbol to construct a many-to-one mapping $\mathcal{B}$ between the extended output symbol set $L \cup \{blank\}$:

$$\mathcal{B}: \mathrm{L} \cup \{\mathrm{blank}\} \mapsto L$$

Then the probability $P(\mathbf{l}\,|\,\mathbf{x})$ can be computed as the accumulated sum of probabilities of all possible alignment paths belong to given target label sequence:

$$P(\boldsymbol{l}|\boldsymbol{x}) = \sum_{\pi\in\mathcal{B}^{-1}(\boldsymbol{l})} P(\pi|\boldsymbol{x}) = \sum_{\pi} \prod_{t=1}^{T} P(\pi_t|\boldsymbol{x}))$$

Due to this sequence modeling ability, CTC is one of the most popular end-to-end (E2E) model for ASR, and has shown competitively performance in acoustic-to-word systems [3][4].

3.2 Attention-based Encoder-decoder

Attention-based encoder-decoder (S2S) [5][16][17][18] is another branch of E2E model. Unlike taking the conditional independent assumption in CTC, it predicts the posterior probability of the label sequence given both the feature sequence x and the previous inference labels $\mathrm{l}_{1:i-1}$.

$$P(\mathbf{l}|\mathbf{x}) = \prod_i P(l_i|\mathbf{x}, \mathbf{l}_{1:i-1})$$

where $\mathrm{P(l_i|x, l_{1:\mathit{i}-1})}$ is obtained by

$$\mathrm{h} = \mathrm{Encoder(x)}$$
$$\alpha_{\mathrm{it}} = \mathrm{Attention}(\mathrm{q}_i, \mathrm{h}_t)$$
$$\mathrm{c}_i = \sum_t \alpha_{\mathrm{it}} \mathbf{h}_t$$
$$\mathrm{P(l_i|x, l_{1:\mathit{i}-1})} = \mathrm{Decoder}(\mathrm{l_{i-1}}, \mathrm{q}_{i-1}, \mathrm{c}_i)$$

Where $\mathrm{Encoder}(\cdot)$ can be LSTM or bidirectional LSTM (BLSTM) and $\mathrm{Decoder}(\cdot)$ can be LSTM or gated recurrent unit (GRU). Since the $\mathrm{Attention}(\cdot)$ calculates the weighted sum of the hidden vectors encoded from feature sequence, it can automatically learn the soft alignment between feature and label sequence.

3.3 Modular Training and Decoding Framework

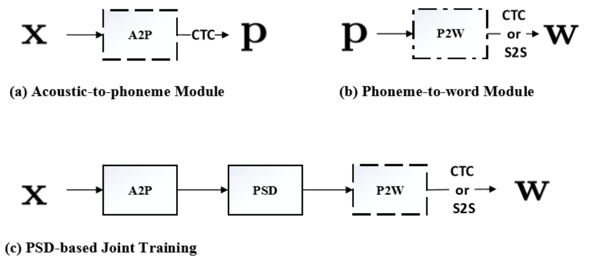
Though both CTC and S2S models can directly build an acoustic-to-word system, they still have fixed vocabulary and cannot handle the out-of-vocabulary (OOV) problem. In addition, the paired acoustic-scripts corpus is needed and a huge amount of text data cannot be utilized directly in these frameworks. Therefore, we use the previously proposed modular acoustic-to-word framework [10] to build our E2E ASR system.

As shown in Figure 1, the E2E word sequence recognition is modularized as an acoustics-to-phoneme model (A2P) and a phoneme-to-word model (P2W). In this paper, A2P is trained by CTC criterion using acoustic data. Meanwhile, P2W is trained by CTC or S2S using text data.

Then modules are integrated into an A2W model by phone synchronous decoding (PSD) [19] and joint optimization.

$$\mathrm{P(w|x)} \approx \max_{\mathrm{p}}[\,\mathrm{P(w|p)} \cdot \mathrm{PSD(\,P(p|x)\,)}\,]$$

Where $\mathbf{w}$, $\mathbf{p}$ and $\mathbf{x}$ are word sequence, phoneme sequence and acoustic feature sequence, respectively.



(a) Acoustic-to-phoneme Module

(b) Phoneme-to-word Module

(c) PSD-based Joint Training

Figure 1: *Framework of modular training of neural acoustics-to-word mode [10]. The solid line box denotes the layers whose parameters are fixed. The dash line and dash-dot line boxes denote that models are trained from acoustic data and text data respectively.*

4. OOV Words extension

As the modularized A2P model expects acoustic input, the text corpora can only be used to improve the P2W part. As shown in Figure 2, we can directly extend the output layer in P2W module to model the desired OOV words. To train these extended OOV words, we need filtering the extra text to increase the relevance, synthesizing the additional P2W data, and fine-tuning the pre-trained P2W model.

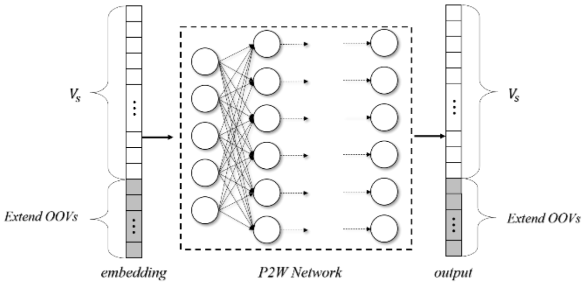


Figure 2: *OOV words extension for P2W module.*

4.1 OOV P2W Data Augmentation

4.1.1 Additional Text Filter

Compared to the fully labeled speech data, extra text data are more easily to be collected. However, it is observed that OOV words are usually low-frequency words and only parts of these additional text data contain useful information. That is why we need to filter the text to increase the pertinence between the training data and OOV.

The simplest way is to select the sentences which contain the OOV words. Considering that OOV words often appear in relatively long sentences, we can also cut the sentence and only use the OOV ngram segments to construct the additional OOV text data. In practice, we tried both strategies and found they almost have the same ability to recover the OOVs. The result showed in the paper utilizes the full sentences for experiments.

4.1.2 Synthetic Inputs

To fine-tune the P2W model by these additional OOV text data, we need to map the word sequence into a phoneme sequence. [20] proposed an appropriate method to generate synthetic input. However, such a method is not suitable for CTC models due to the spiky predictions. We found that the phoneme sequences showed no statistical significance after PSD, so we randomly repeated phonemes one or two times and inserted the *blank* symbol zero to two times to simulate PSD results of the front-end CTC A2P outputs.

What's more, inspired by label smoothing [21], we can smooth the one-hot phoneme for the robust training[2]. The example can be seen in Table 1. We found that the smoothed CTC-PSD-phonestream will get the best performance.

Table 1: *Examples of word "collision" under different synthetic input generation schemes.*

Synthetic Input	Example Sequence
Word	`collision`
Phonestream	`k ax l ih zh ih n`
CTC-PSD-phonestream	`k ax ax blank blank l ih blank zh ih ih n n`

4.2 OOV Data Fine-tuning

Let $\mathcal{D}$ be the ASR dataset, with phoneme distribution input after A2P and PSD and word sequence outputs pairs $(\mathrm{x}_j, \mathrm{y}_j)$ where $\mathrm{j} \in \{1,2,\cdots,|\mathcal{D}|\}$. Using the filtered text corpora and one of the proposed synthetic input creation schemes, we can get the OOV augmentation dataset $\mathcal{A}$, which is comprised of synthetic data pairs $(\widetilde{\mathrm{x}_\mathrm{k}}, \widetilde{\mathrm{y}_\mathrm{k}})$ where $\mathrm{k} \in \{1,2,\cdots,|\mathcal{A}|\}$. We should note that OOV words are usually low-frequency words. Even though we synthesize the OOV P2W data from the extra text corpus, the filtered oov data is still relatively smaller, so generally, we have $|\mathcal{A}| < |\mathcal{D}|$. In all fine-tuning experiments, we always evaluate our model on a held-out ASR dataset $\mathcal{D}'$.

To utilize the augment dataset $\mathcal{A}$, we proposed three types of fine-tune training schemes as below:

1. *Directly fine-tune*: we adopt a small learning rate (in practice, we used 1e-5), and just only use the dataset $\mathcal{A}$ to fine-tune the P2W part of the pre-trained modular system.
2. *Alternative train*: during P2W fine-tune training, we alternate between epochs from the acoustic dataset $\mathcal{D}$ and the augmentation dataset $\mathcal{A}$.
3. *Multi-Modal*: this scheme is only adopted in S2S P2W module. Like [20], we use two encoders and one decoder to construct the S2S P2W model (shown as Figure 3). The dataset $\mathcal{D}$ and $\mathcal{A}$ are mixed together but alternatively fed into different encoders in batches during the training.

The alternative training is different from multi-modal training which $\mathcal{D}$ and $\mathcal{A}$ shared the same encoder in S2S.

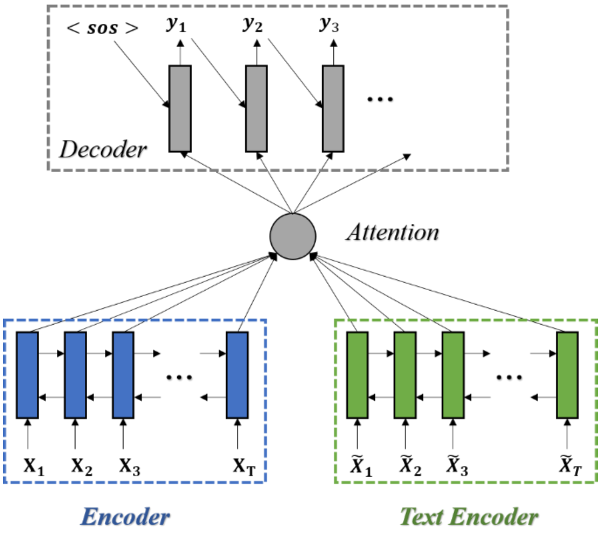


Figure 3: *Multi-Modal S2S model*

5. Experiments

5.1 Experimental Setup

The main data corpus used for the experiments is Switchboard (SWBD) corpus [22]. This corpus contains about 300 of hours speech. The 2000 hours Fisher transcripts and WSJ transcripts were used as the extra text corpus. The evaluation was carried out

[2] smoothing one-hot example: $[0,1,0] \longrightarrow [0.05, 0.9, 0.05]$

on the NIST Eval2000 CTS test set for in-domain experiments and WSJ dev93 data set for cross-domain experiments. 36-dimensional filterbank over 25 ms frames with 10 ms frame shift was extracted as acoustic features. Neural networks were trained by MXNet [23], Kaldi [24] and EESEN [25].

We took the phoneme CTC as the A2P module. The CTC A2P model units are 45 monophones with a *blank*. The baseline CTC A2P was a network with 5-layer LSTMs, and each layer has 1024 memory cells and 256 projection nodes [1]. The P2W module has two versions. The CTC P2W version was a network with 5-layer bidirectional LSTMs, and each layer contains 512 memory cells per direction. The S2S P2W version contains an encoder with 3-layer bidirectional LSTMs and a decoder with 3-layer LSTMs. Each layer of both encoder and decoder networks has 300 memory cells. Dot product attention mechanism was adopted for fewer parameters and faster training.

The size of full vocabulary $\mathcal{V}_f$ was 30K as the standard evaluation setup in SWBD corpus. Lots of these words occur less frequently in training data. In order to stress the OOV problem, we only predicted the words which occur more than 10 times in training set, resulting in a small vocabulary $\mathcal{V}_s$ with size 6805. Here in-vocabulary (IV) was defined as words in $\mathcal{V}_s$ and the out-of-vocabulary (OOV) meant word not in $\mathcal{V}_s$. During P2W training, the OOV words in train labels were replaced by a special symbol <*unk*>. The 30k and 6.8k vocabulary size P2W model were our two baseline systems.

Word error rate (WER) was taken as the metric. To investigate on the OOV WER gain from the proposed method, we split the test dataset into two categories depending on whether all words appear in $\mathcal{V}_s$ or not, referred as in-vocabulary sentences (IVS) and out-of-vocabulary sentences (OOVS) respectively.

In the remaining of the paper, if not explicitly stated, we always adopt the proposed phoneme CTC as our A2P module.

5.2 OOV Extension on In-domain Eval2000

To extend the OOV words in Eval2000, we calculated the Eval2000 test set vocabulary $\mathcal{V}^{eval2000}$. The OOV word vocabulary should be $\mathcal{V}_{oov}^{eval2000} = \mathcal{V}^{eval2000} - \mathcal{V}_s$, with size 843, and the final extended vocabulary was $\mathcal{V}_{extend}^{eval2000} = \mathcal{V}_{oov}^{eval2000} + \mathcal{V}_s$, with size 7648. According to the words in $\mathcal{V}_{oov}^{eval2000}$, we can get the extra text data set $\mathcal{A}^{eval2000}$ (augmented dataset, has about 64882 utterances) in Fisher corpus. The vocabulary and dataset information are shown in Table 2. In practice, we didn't extend the interjections like *mhm*, so the test OOV rate is not zero.

Table 2: *Vocabulary Information*

Vocabulary	Size	OOV Rate		
		SWBD Train	Eval2000 Test	WSJ Dev93 Test
$\mathcal{V}_f$	30275	0	1.47	6.4
$\mathcal{V}_s$	6805	2.04	3.33	15.2
$\mathcal{V}_{\text{extend}}^{\text{eval2000}}$	7648		0.27	-
$\mathcal{V}_{\text{extend}}^{\text{dev93}}$	7627		-	1.2

Table 3 shows the performance comparison among baseline systems and proposed OOV extension systems on the in-domain test set. The 30k and 6.8k baseline systems show that specific filtering on the vocabulary may not hurt the system performance, and even can improve results by ignoring the low-frequency words.

It also shows using the augmented text data to directly fine-tune the pre-trained P2W module does not work well. It is because the augmented data is mismatch with the real A2P module output. Only using the augmented data may mislead the P2W module. Multi-modal S2S P2W module gives slight improvements while alternative training can significantly improve the performance of the P2W module on OOV sentences. These results indicate the augmented OOV text data can recover the OOV words without hurting the IV sentences WER.

Table 3: *WER (%) comparison on Eval2000 with OOV Extension Fine-tune Training*

System	P2W	Output	WER		
			All	IVS	OOVS
Mod. S2S	S2S	30275	27.7	24.8	33.0
		6805	27.9	25.2	33.1
+ directly fine-tune	S2S	7648	30.9	28.3	35.8
+ alternative train			**26.9**	**24.7**	**31.2**
+ multi-modal			27.5	24.7	32.7
Mod. CTC	CTC	30275	30.2	29.5	31.5
		6805	25.8	23.4	30.3
+ directly fine-tune	CTC	7648	26.3	23.8	31.0
+ alternative train			**25.0**	**23.1**	**28.7**

5.3 OOV Extension on Cross-domain WSJ Dev93

WSJ is a well-known English clean speech database [26][27]. We adopt the WSJ dev93 part as a cross-domain test set for OOV extension experiments.

Like eval2000 dataset, we calculated the dev93 test set vocabulary to get the oov vocabulary $\mathcal{V}_{oov}^{dev93}$, and added the small vocabulary set $\mathcal{V}_s$ to get the dev93 extended vocabulary $\mathcal{V}_{extend}^{dev93}$. The details of it can be seen in Table 2.

Since the training set (SWBD) and test set (WSJ) are in a different domain, the vocabulary between them have a big difference. The small vocabulary selected from SWBD training set $\mathcal{V}_s$ gets 15.2% oov rate on dev93 test set, resulting in high WER. We used the extra text data set $\mathcal{A}^{dev93}$ (augmented dataset, has about 60297 utterances) from the WSJ transcripts to extend the OOV words.

Table 4 demonstrates the performance comparison among baseline systems and proposed OOV extension systems on the cross-domain test set. As we do not use any acoustic data of WSJ, the WERs showed in Table 4 on dev93 are relatively higher than any other reported results. It is observed that directly fine-tune also performs bad and multi-modal is not suited to cross-domain problems. However, the alternative train method slightly improves the performance of IVS and significantly reduces the WER on OOVS. It is believed that the modular E2E system with our OOV extension method can simply and effectively handle the OOV problems across the domain.

Table 4: *WER (%) comparison on WSJ dev93 with OOV Extension Fine-tune Training*

System	P2W	Output	WER		
			All	IVS	OOVS
Mod. S2S	S2S	30275	43.81	22.63	46.53
		6805	41.32	20.42	44.01
+ directly fine-tune	S2S	7627	39.17	20.53	42.27
+ alternative train			**35.65**	**18.95**	**37.80**
+ multi-modal			40.76	18.53	43.62
Mod. CTC	CTC	30275	39.19	26.21	40.86
		6805	36.47	18.42	38.79
+ directly fine-tune	CTC	7627	36.92	17.68	39.40
+ alternative train			**30.33**	**17.89**	**31.93**

6. Conclusion and Future Work

In this paper, we proposed a method to extend the OOV words in E2E ASR system. Adopt modular E2E ASR framework and finetune the P2W module with extra text data. This novel design gives the system an easy way to extend the output and directly utilize the text data. It shows a slight improvement on in-domain test set and a large gain on the cross-domain OOV problem. To further investigate, our future work will concentrate more on cross-domain and domain adaptation problems.

Acknowledge

Yanmin Qian and Kai Yu are the corresponding authors. This work has been supported by the National Key Research and Development Program of China under Grant No.2017YFB1002102, China NSFC project (No. 61603252) and China NSFC project (No. U1736202). Experiments have been carried out on the PI supercomputer at Shanghai Jiao Tong University.

Reference

[1] Sak H, Senior A, Beaufays F. Long short-term memory recurrent neural network architectures for large scale acoustic modeling[C]//Fifteenth annual conference of the international speech communication association. 2014.

[2] Chen Z, Droppo J, Li J, et al. Progressive joint modeling in unsupervised single-channel overlapped speech recognition[J]. IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP), 2018, 26(1): 184-196.

[3] Audhkhasi K, Kingsbury B, Ramabhadran B, et al. Building competitive direct acoustics-to-word models for english conversational speech recognition[C]//2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018: 4759-4763.

[4] Yu C, Zhang C, Weng C, et al. A multistage training framework for acoustic-to-word model[J]. ISCA INTERSPEECH, 2018.

[5] Weng C, Cui J, Wang G, et al. Improving attention-based sequence-to-sequence models for end-to-end english conversational speech recognition[C]//Interspeech 2018. 2018.

[6] Zeyer A, Irie K, Schlüter R, et al. Improved training of end-to-end attention models for speech recognition[J]. arXiv preprint arXiv:1805.03294, 2018.

[7] Sennrich R, Haddow B, Birch A. Neural machine translation of rare words with subword units[J]. arXiv preprint arXiv:1508.07909, 2015.

[8] Palaskar S, Metze F. Acoustic-to-word recognition with sequence-to-sequence models[C]//2018 IEEE Spoken Language Technology Workshop (SLT). IEEE, 2018: 397-404.

[9] Li J, Ye G, Zhao R, et al. Acoustic-to-word model without OOV[C]//2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU). IEEE, 2017: 111-117.

[10] Chen Z, Liu Q, Li H, et al. On modular training of neural acoustics-to-word model for lvcsr[C]//2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018: 4754-4758.

[11] Bazzi I. Modelling out-of-vocabulary words for robust speech recognition[D]. Massachusetts Institute of Technology, 2002.

[12] Bisani M, Ney H. Open vocabulary speech recognition with flat hybrid models[C]//Ninth European Conference on Speech Communication and Technology. 2005.

[13] Chen Z, Zhuang Y, Yu K. Confidence measures for ctc-based phone synchronous decoding[C]//2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2017: 4850-4854.

[14] Hayashi T, Watanabe S, Zhang Y, et al. Back-translation-style data augmentation for end-to-end ASR[C]//2018 IEEE Spoken Language Technology Workshop (SLT). IEEE, 2018: 426-433.

[15] Graves A, Fernández S, Gomez F, et al. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks[C]//Proceedings of the 23rd international conference on Machine learning. ACM, 2006: 369-376.

[16] W. Chan, "End-to-end speech recognition models," Ph.D. dissertation, Carnegie Mellon University Pittsburgh, PA, 2016.

[17] E. Variani, T. Bagby, E. McDermott, and M. Bacchiani, "End-to-end training of acoustic models for large vocabulary continuous speech recognition with tensorflow," Proc. Interspeech 2017, pp. 1641–1645, 2017.

[18] R. Prabhavalkar, K. Rao, T. N. Sainath, B. Li, L. Johnson, and N. Jaitly, "A comparison of sequence-to-sequence models for speech recognition," Proc. Interspeech 2017, pp. 939–943, 2017.

[19] Z. Chen, Y. Zhuang, Y. Qian, and K. Yu, "Phone synchronous speech recognition with ctc lattices," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 25, no. 1, pp. 86– 97, Jan 2017.

[20] A. Renduchintala, S. Ding, M. Wiesner, and S. Watanabe, "Multi-modal data augmentation for end-to-end asr," in Proc. Interspeech 2018, 2018, pp. 2394–2398. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-2456

[21] G. Pereyra, G. Tucker, J. Chorowski, L. Kaiser, and G. Hinton, "Regularizing neural networks by penalizing confident output distributions," arXiv preprint arXiv:1701.06548, 2017.

[22] J. J. Godfrey, E. C. Holliman, and J. McDaniel, "Switchboard: Telephone speech corpus for research and development," in Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on, vol. 1. IEEE, 1992, pp. 517–520.

[23] T. Chen, M. Li, Y. Li, M. Lin, N. Wang, M. Wang, T. Xiao, B. Xu, C. Zhang, and Z. Zhang, "Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems," arXiv preprint arXiv:1512.01274, 2015.

[24] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz et al., "The kaldi speech recognition toolkit," in IEEE 2011 workshop on automatic speech recognition and understanding, no. EPFLCONF-192584. IEEE Signal Processing Society, 2011.

[25] Miao Y, Gowayyed M, Metze F. EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding[C]//2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU). IEEE, 2015: 167-174.

[26] J. Garofalo, D. Graff, D. Paul, and D. Pallett, "Csr-i (wsj0) complete," Linguistic Data Consortium, Philadelphia, 2007.

[27] LD Consortium. CSR-II (wsj1) complete[J]. Linguistic Data Consortium, Philadelphia, vol. LDC94S13A, 1994.